Callis Author User's Guide

Applicable from Callis Author version 3.3

Document version 3.0

This document describes how to use Callis Author.

Contents

1 Callis Author overview 4

1.1 Callis Author components 4

1.2 An example of a typical use scenario 4

1.3 The content of this document – where to find which information 5

2 Structuring process descriptions 6

2.1 Common structure of description elements 6

2.2 Description elements for documenting work methods 6

2.3 Description elements for documenting work flows 7

3 Managing standard processes (CA - Process Control Center) 9

3.1 Overview of and maneuvering the Process Control Center 9

3.2 Defining and organizing standard processes 10

3.3 Defining and organizing packages 11

3.3.1 Why should I organize my process in folders with packages? 11

3.3.2 Which packages should I have? 11

3.3.3 How do I manage folders and packages? 12

3.4 Authoring a standard process 13

3.5 Publish a standard process 15

4 Modeling and authoring processes (CA - Editor) 16

4.1 Components of the editor 16

4.1.1 The views in the accordions 16

4.1.2 The element editor 16

4.1.3 The task step editor 18

4.1.4 The external editor 20

4.2 Versioning, approving, and archiving description elements 22

4.2.1 Versioning mechanism 22

4.2.2 Approval mechanism 22
4.2.3 Archiving mechanism 23
4.3 Editing html blocks 23
4.3.1 Creating links in text blocks 23
4.3.2 Embed clickable images into html fields 23
4.3.3 What subset of HTML tags are allowed in the Online Editor 23
4.4 Working with pictures 24
4.4.1 How do I make links from the Visio overview pictures to description elements? 24
4.4.2 How do I insert a Visio picture in an taskgroup or guidance element 24
4.4.3 The gif files exported by Visio have a very low resolution 25
4.5 Configuring the publishing properties 25
4.6 Publishing the process 26
5 Browsing a standard process 27
5.1 Layout of Callis Author - Presenter 27
5.2 Which views exists in the Presenter and how are they generated? 27
5.2.1 Workflows view 27
5.2.2 Disciplines view 28
5.2.3 Types view 28
5.2.4 Standards & Models view 28
5.3 Change / comparison information 28
6 About the reports 29
6.1 Entire Process as PDF (static, updated when building or publishing) 30
6.2 PIID Report template 30
6.3 Statistics 30
6.4 Process Integrity 30
6.5 Coverage against Standards 31
6.6 Process Overview 31
6.7 Who references [current] 31
6.8 Unresolved References 31
6.9 Element versions – Grouped by package 31
6.10 Element versions – Grouped by save date 32
6.11 Elements compared to other process 32
7 Comments 33

7.1 Commenting on published process descriptions 33
7.1.1 Do a comment 33
7.1.2 Get an overview of recently changed comments 34
7.1.3 See comments across a set of projects 34
7.1.4 Migrate comments across process versions 34

1 Callis Author overview

Callis Author is a web based application for **modeling**, **documenting**, **tailoring**, **publishing** and **improving** business processes.

1.1 Callis Author components

Callis Author runs in a web server and consists of the following overall components:

- **Callis Author - Process Control Center (PCC)** is used to
  - Create *packages* with process content (description elements)
  - Combine packages into *standard processes*
  - Open a standard process for authoring
  - Publish a standard process
- **Callis Author - Editor** is used to
  - Author and model processes
  - Edit content of description elements (to create process content, e.g. task and role descriptions)
  - Manage compliance against standards and requirements
- **Callis Author – Presenter** is used to
  - Present and make available the standard process to process users using multiple views and effective search
  - Make a near real-time version of the current edited standard process available as it will look to the end users to make it easier for authors to create high-quality and easy understandable process descriptions.
- **Callis Author – Administration** is used to
  - Define projects

Depending on the settings of your browser these components will open either as new tabs or windows.

1.2 An example of a typical use scenario

A typical use scenario would be the following:

1. The **Process Manager** (overall responsible for the processes in the organization) decides how to organize process description into packages (typically done based on who should be responsible for which groups of processes). Examples are packages like “requirements management”, “project planning and tracking”, and “test management”. The Process Manager uses Callis Author – Process Control Center to create the relevant packages and combine these into a standard process. The standard process is opened for authoring and the process authors are asked to write process content.
2. The **Process Author(s)** start modeling and authoring the process descriptions using *Callis Author – Editor.* Immediately, the Process Authors can see the resulting standard process using *Callis Author – Presenter*.

3. When the Process Authors are done modeling, the **Process Manager** controls the quality of the standard process using the build-reports for checking process integrity. Processes can be approved and when ready, the standard process is published using Callis Author - PCC.

Please notice that this is just one example of how to use Callis Author. Callis Author supports a number of different usage scenarios, ranging from one user writing and managing all process descriptions to multiple process authors, authoring and managing multiple process descriptions and projects or organizational units tailoring these descriptions to fit their exact need.

1.3 The content of this document – where to find which information

Chapter	Title	Explains how to…
2	Structuring process descriptions	• model processes using Callis Author description elements
3	Managing standard processes (CA - Process Control Center)	• combine packages with process content into processes • open a process for authoring • publishing a process
4	Modeling and authoring processes (CA - Editor)	• author process descriptions
5	Browsing a standard process	• browse the standard process
6	About the reports	• use the different reports to improve and get overview of the process

2 Structuring process descriptions

Callis Author defines a number of description elements. Examples of description elements are role, task, artifact, guideline, and checklist.

When using Callis Author to produce process descriptions, instances of description elements are created. An example of an instance of the role description elements could be “Project Manager”. Examples of instances of tasks could be “Define work breakdown structure” and “Release component for integration”.

2.1 Common structure of description elements

All description elements have an ID (called ***nid)*** and a ***name***. The *nid* is uniquely identifying the description element and is used when referencing the element and making relations between elements, e.g. including a specific task in a specific task group. The *name* is prompted for, when creating a new instance of a description element. A suggestion for a *nid* is generated by Callis Author based on the *name* of the instance.

Furthermore, all description elements have a *summary block*, a *description block* and a *more information block*.

The ***summary block*** is used when generating and presenting short overview information about the element in lists, tool tips etc.

The ***description block*** is intended to be the place where the general information about the element is documented.

The ***more information*** *block* is used to refer to elements of the type “Guidance”, i.e. checklist, concept, example etc.

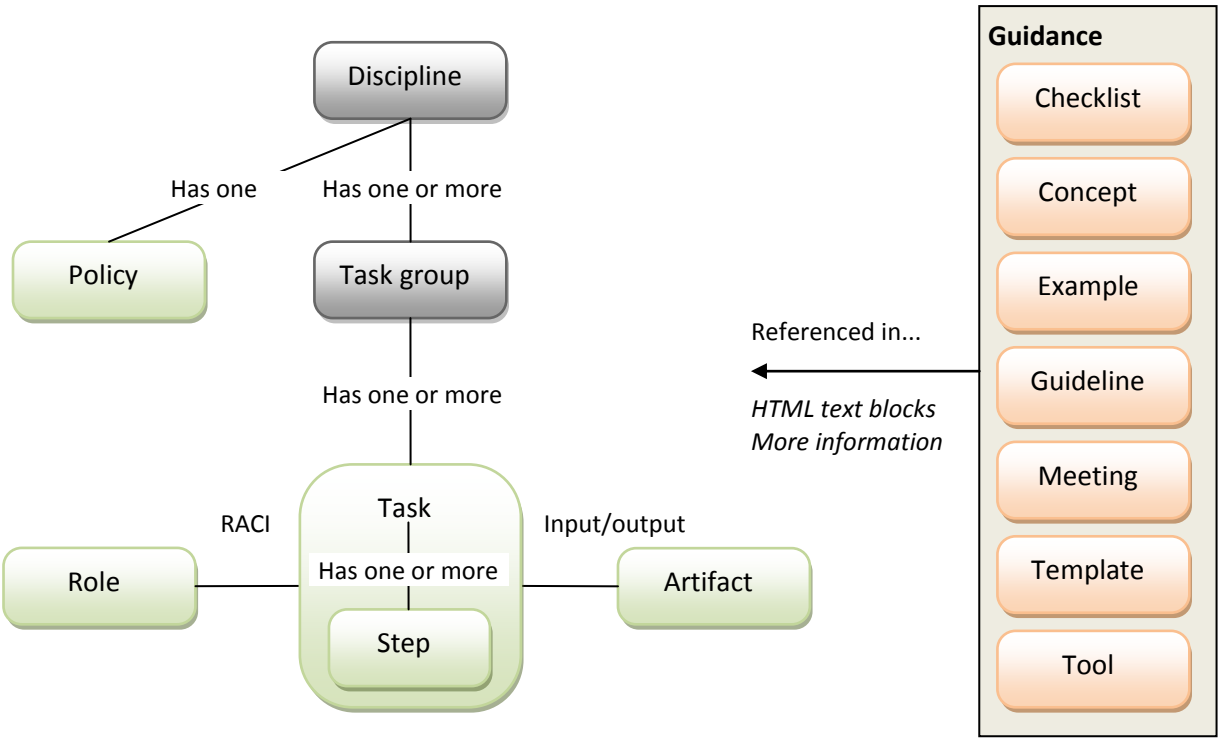
Besides the common structure of all description elements, the some elements have specialized blocks, e.g. the disciplines have a block for referring to a policy, and the tasks have a number of steps.

2.2 Description elements for documenting work methods

The most important description elements are:

- **Role** is used to describe roles. A role will be responsible performing specific tasks. Examples of often seen instances of the role element are: “Project Manager”, “Requirements Manager”, “Architect”, and “Developer”.
- **Task** is used to describe units of work. The task description element can define a list of steps, which is useful when documenting detailed work processes. Examples of tasks could be “Define work breakdown structure” and “Release component for integration”. Normally, when producing process descriptions using Callis Author, tasks would be the description elements which is used the most.
- **Artifact** is used to describe work products. Examples of often seen artifacts within Software Engineering are “Requirements Specification”, “Test Case”, and “Project Schedule”.

The “Role”, “Task”, and “Artifact” description elements are the most important elements when describing work methods. The description elements and their relations are illustrated on the following picture.



Picture 1 shows the description elements and relations used when modeling work methods.

Furthermore, Callis Author defines a set of “Guidance” elements, which are useful to support the descriptions of work methods. These elements are illustrated to the right in the picture.

The “Policy” element is used to define a policy within a specific discipline.

The “Discipline” and “Task Group” elements (white on the illustration) are used to group related task, e.g. tasks related to “Configuration Management”. This is useful to get an overview of tasks related to a specific discipline and when generating different views of the content of the process.

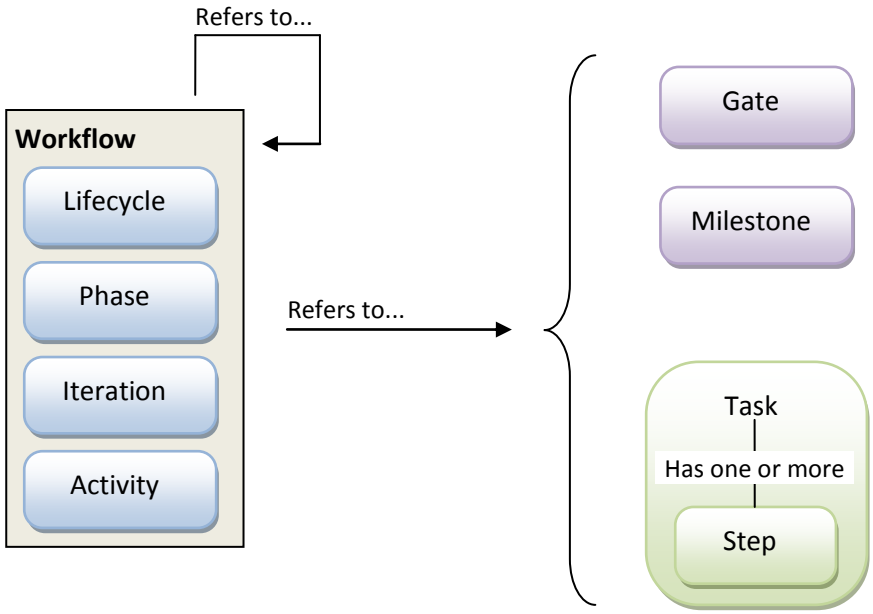
The domain model in Callis Author defines a number of possible relations between description elements. These relations are used to create different views on the standard process and to create validating the standard process, e.g. generating “Integrity Report” and “Process Overview”.

2.3 Description elements for documenting work flows

Callis Author supports creation of process oriented work flows on top of the work methods presented in the previous section. This is useful when documenting e.g. overviews of company processes, and cross-functional workflows, e.g. swim lanes.

Please notice, that the task elements are bridging the process oriented work flows and the work methods.

The following picture illustrates the work flow related description elements and their relations.



Picture 2 shows the description elements and their relations used when documenting workflows.

The **Lifecycle, Phase, Iteration, and Activity** description elements are used to document work flows. Examples are model of complete business, phase/gate models, swim lane, etc. These work flow elements can refer other work flow elements and gates, milestones, and tasks, which is useful when building up hierarchical views of processes.

The **Gate** and **Milestone** description elements can be used to document milestones and gates.

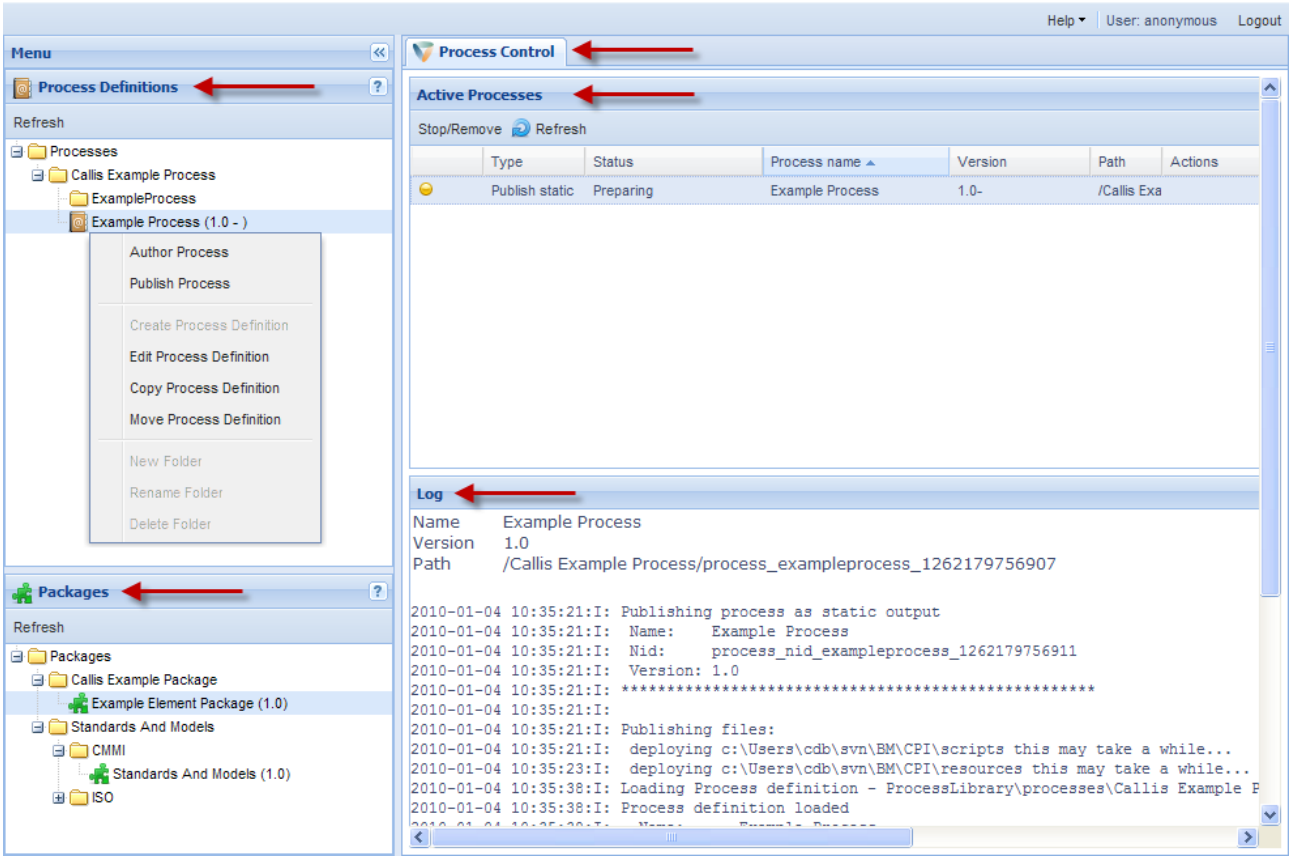
3 Managing standard processes (CA - Process Control Center)

3.1 Overview of and maneuvering the Process Control Center

The Callis Author – Process Control Center consists of the following components

- A menu with two accordions:
  - "Process Definitions" contains a tree with folders and nodes (= process definitions)
  - "Packages" contains a tree with folders and nodes (= package definitions)
- A tab "Process Control" with two windows:
  - Active Processes contains a table with references to process definitions open for authoring or published
  - Log shows the output of compilation of a specific process definition

The components are marked with red arrows in the following screen shot.



You can right-click the tree nodes in the accordions to execute commands on packages and process definitions. In the screen shot the user has right-clicked on the process definition called "Example Process (1.0 -)", and a menu has appeared.

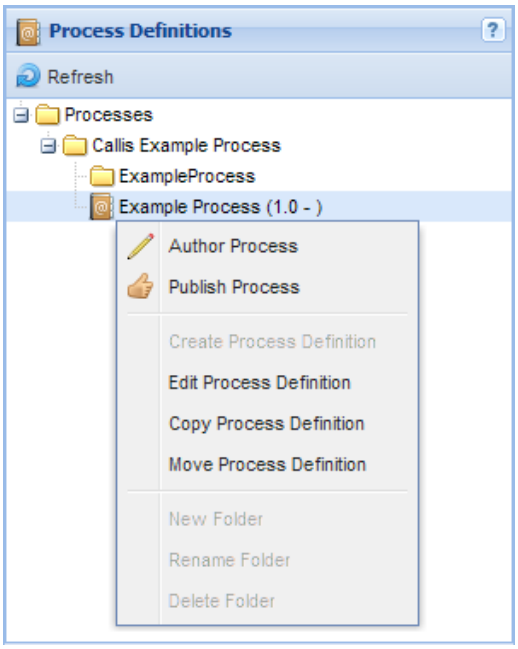
Furthermore, **you can double-click process definition nodes and package nodes** to open these for editing.

3.2 Defining and organizing standard processes

Process definitions are organized in a hierarchy of folders. The folders are mapped to physical folders on disk. A process definition can be moved to another folder. This will not break the process definition.

The following operations can be performed by right clicking on a node in the process definition tree in the accordion:

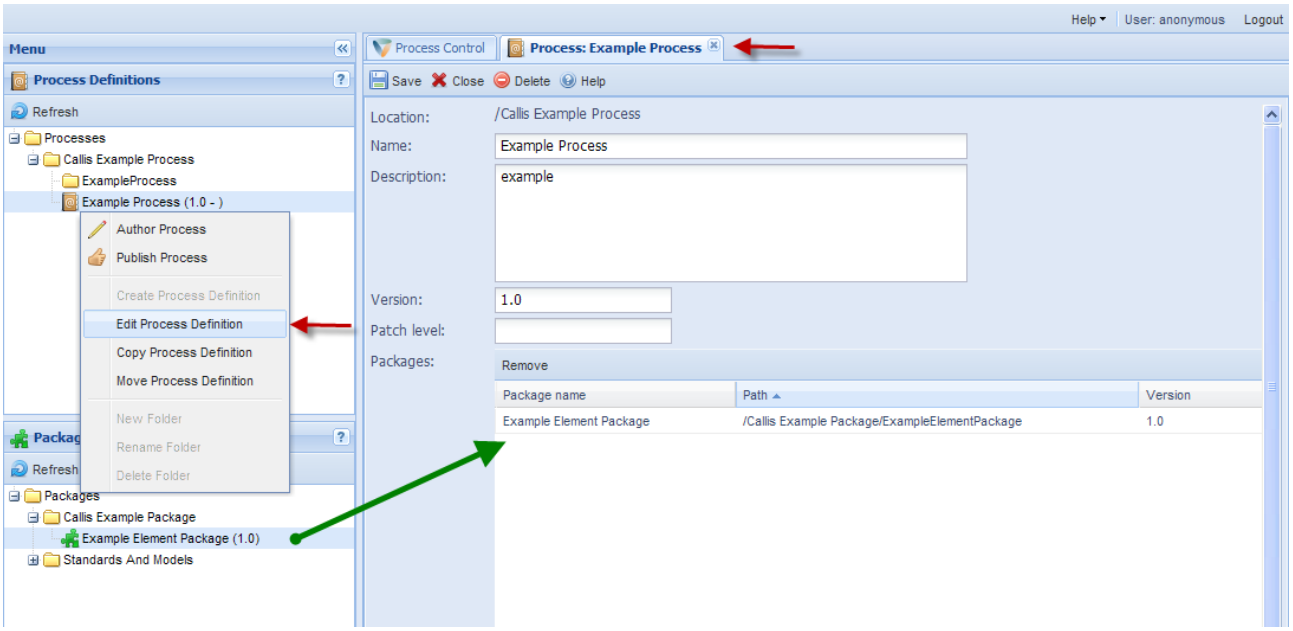
- **Author** or **publish** the process
- **Create**, **Edit**, **Copy**, or **Move** the process definition
- **Create**, **Rename**, or **Delete** folders to hold process definitions



Please notice that some of the operations are allowed on folders and others on process definitions. This is reflected by dimming the entries in the menu when right-clicking in the process definition accordion.

Furthermore, process definitions can be opened for editing by double-clicking on the process definition icon. This will open a tab with the process definition data.

When the process definition is edited, a tab appears in the main window (marked with red arrows on the following screen shot):



Name, description, version, and patch level can be edited. Furthermore, the content of the process can be defined by dragging references to packages from the packages accordion to the "Packages" table in the tab (marked with a green arrow in the screen shot). Please notice that in this example only a single package is referenced, but this can be done with multiple packages.

3.3 Defining and organizing packages

Packages contain process content (description elements) and are organized in a hierarchy of folders.

3.3.1 Why should I organize my process in folders with packages?

There are a number of reasons for organizing you process description into folders and packages:

1. You can use the organization in packages to help you build standard processes that fit the use of e.g. specific types of projects in specific domains, as packages are the "building blocks" of standard processes (high level tailoring).
2. At a later point we expect to make a "process owner" delegation mechanism at the package level meaning that changing the process description will require specific access rights to that package.
3. At a later point we expect to support a "package locking and versioning" concept to further support distribution responsibility of process description development and maintenance.
4. It's easier to locate specific description element XML data as the data files are stored below the folder corresponding to the relevant package. As standard process often consists of 100's of description elements, it is nice to have the files organized.

3.3.2 Which packages should I have?

That depends, but in general we recommend that you use the following scheme (you can do it as you like, please report back to us if you have practical experiences to share).

1. Create one package for global role definitions used across multiple packages (typically examples are: Project Leader, Architect, Configuration Manager, and Developer)
2. Create one package for each of you disciplines (typically examples are: Configuration Management, Requirements Management, and Project Management). Examples of typical description elements to go into this package would be: Tasks, artifacts, and various guidance elements.
3. Create one or more packages to describe the work flows of the standard processes. Typically, these go across multiple disciplines.

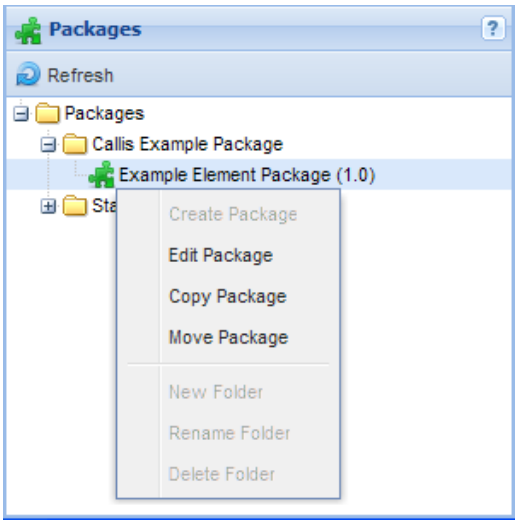
This way of organizing the description elements into packages will support that work flows (or a part of a work flow) and disciplines are changed to support process variations (of course with some limitations).

3.3.3 How do I manage folders and packages?

The folders are mapped to physical folders on disk. Folders can be created and moved by right clicking on the folders. Packages can be created, copied and moved.

The following operations can be performed by right clicking on a node in the package tree in the accordion:

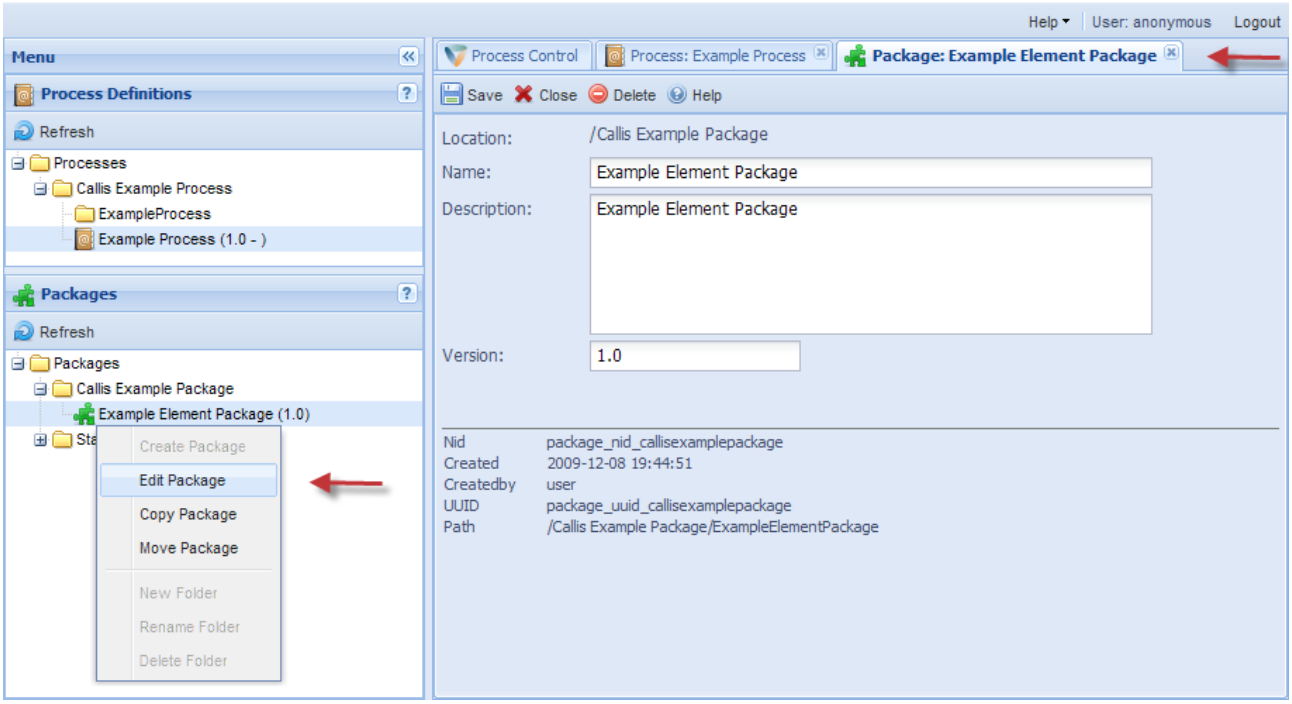
- **Create**, **Edit**, **Copy**, or **Move** the package
- **Create**, **Rename**, or **Delete** folders to hold packages



Please notice that some of the operations are allowed on folders and others on packages. This will be reflected by dimming the entries in the menu when right-clicking in the package accordion.

Furthermore, packages can be opened for editing by double-clicking on the package icon. This will open a tab with the package data as shown in the following screen shot.

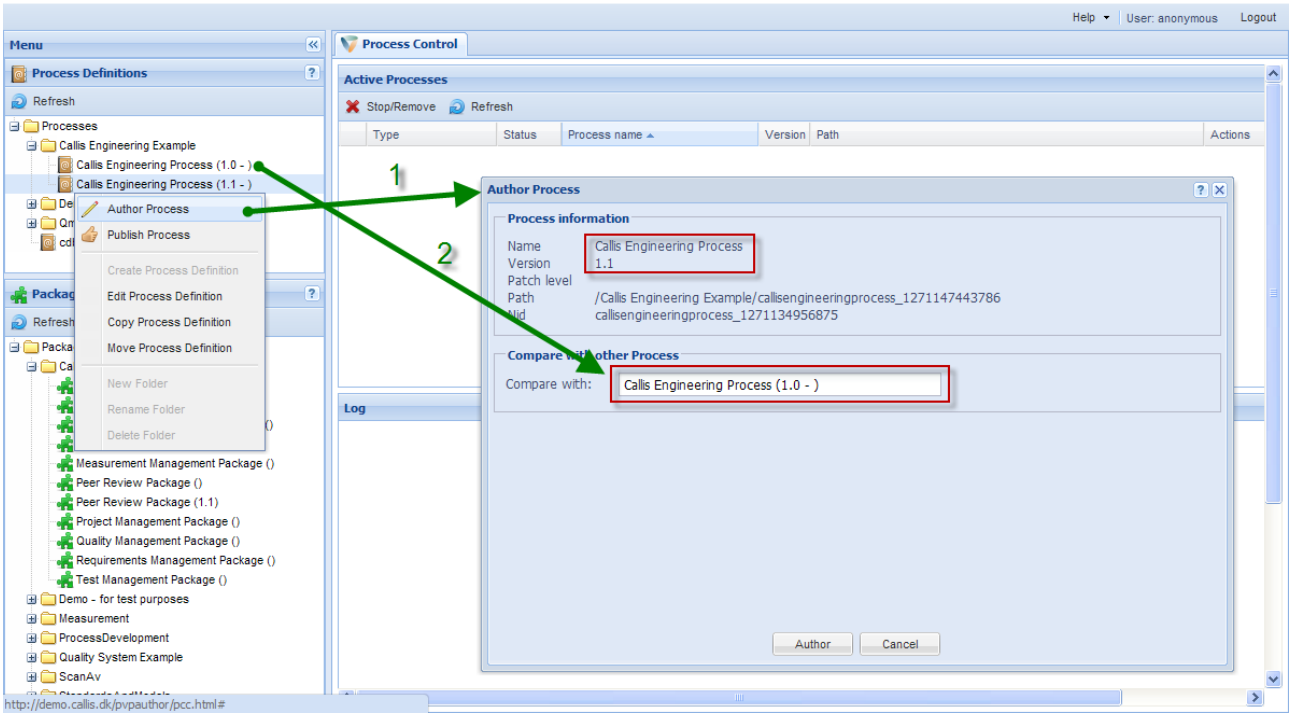
Please notice, if the hierarchy of packages is changed, the process definitions including the moved packages will no longer work as these cannot be automatically updated. If you need to change the package hierarchy or move a folder, you have to open the process definitions using the packages, remove the old package references and insert new package references in the process definitions.



Name, description, and version can be edited. Please notice that the process content of a package is edited by selecting “author process” on a process definition which refers to the relevant package.

3.4 Authoring a standard process

A process definition can be compiled by right-clicking the specific process definition in the Process Definition window to the left. See the following screen shot.



Start the process for authoring by:

1. right-click on the relevant process definition in the Process Definitions window,
2. a menu appears,
3. select “Author” in the menu, and
4. the “Author Process” window appears. Please see the following screen shot where this is illustrated by arrow 1.

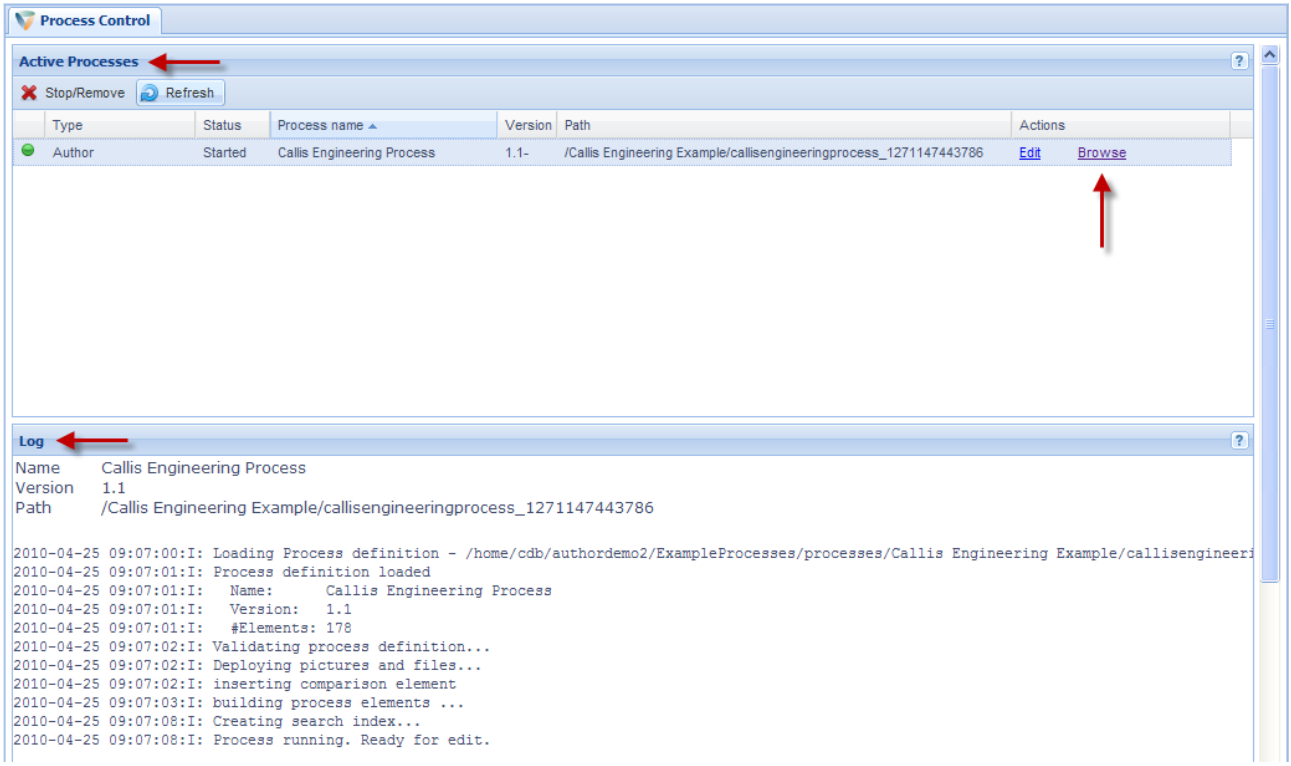
The “Author Process” window consists of two groups of information:

- In “Process Information” you can read information about the process you are about to author including name and version (marked with a red square in the screen shot).
- In the “Compare with other Process” there is an input field where you can drag another process definition from the “Process Definitions” tree to the left (Arrow 2 in the screen shot).

You can choose either to author a process without compare information (then just click the “Author” button without dragging another process definition to compare with) or you can author with compare information. If you are building in compare information, this will result in the change elements in the process being marked and a compare report being generated (please refer to 6.11 for more information about this report).

Please notice that the compare information build into the process will be invalid as soon as you start changing the process (it is calculated at start-up and not maintained when authoring).

When having selected the “Author” button, the “Author Process” window will close and the Process Control tab will be updated.



This tab consists of two windows:

- Active Processes
- Log

A row will appear in the Active Processes window representing the started process definition.

When a row in the Active Processes window is selected, the Log window displays the status of the compilation/start-up of the process. As long as the status icon is yellow, the compilation is not finished. Depending on the result of the compilation, the status icon will change to either red or green.

When the process is ready for authoring, the status icon on the row changes from yellow to green and two links (*Edit* and *Browse*) will appear in the Actions column. Clicking these links will open tabs with Callis Author Editor(for editing) and Presenter (for browsing), respectively. Please notice, that normally it is most convenient to select *Browse*, find the relevant element, and click edit directly from Callis Presenter as this will open the editor with the relevant element preloaded.

If a blocking error appears during compilation, the icon will change from yellow to red. If this occurs, you can read the error text in the Log window. Furthermore, please consult your local expert on Callis Author. More information may be found in the Callis Author Log file.

3.5 Publish a standard process

You can publish you process by right-clicking on the process definition and selecting publish. Similar to when authoring the process, you can choose to compare the process with another process definition or just publish the process without comparing it to another process definition. This works in the exact same way.

The published process will be available as a set of static files and/or pdf-files. This complete set of files can be copied to a standard web server, where the process content can be accessed by process users. You can select to browse your published process by using the browse link provided in the Active Processes window.

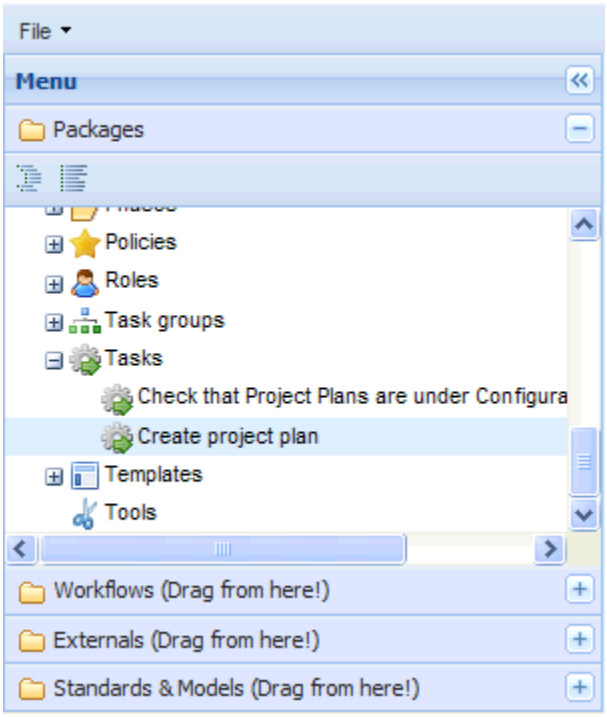
4 Modeling and authoring processes (CA - Editor)

4.1 Components of the editor

4.1.1 The views in the accordions

The editor presents the description elements of the currently edited standard process in the following accordions:

- **Packages view** lists *packages -> types of description elements -> instances of description elements*. From this view new description elements can be created. This view is the most important view when editing elements.
- **Workflows view** lists workflow elements currently defined and their relations
- **Standards & Models view** lists the standards and models currently compiled into the standard process.

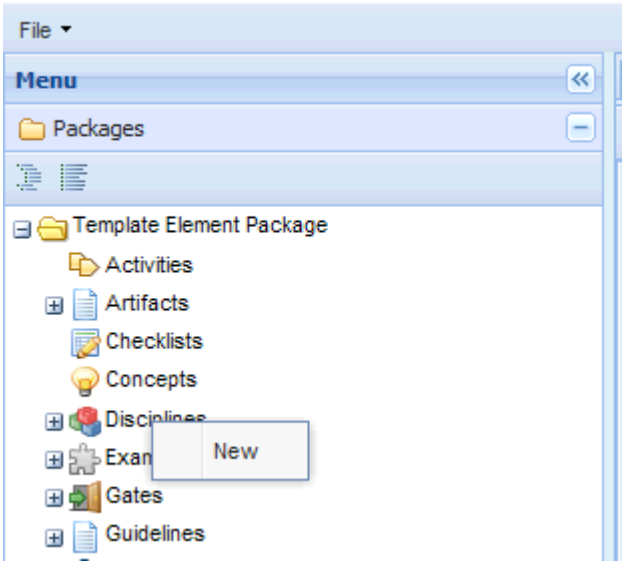


When double clicking a specific description element in the Package view, Callis Author opens a tab making it possible to edit the content of the description element.

4.1.2 The element editor

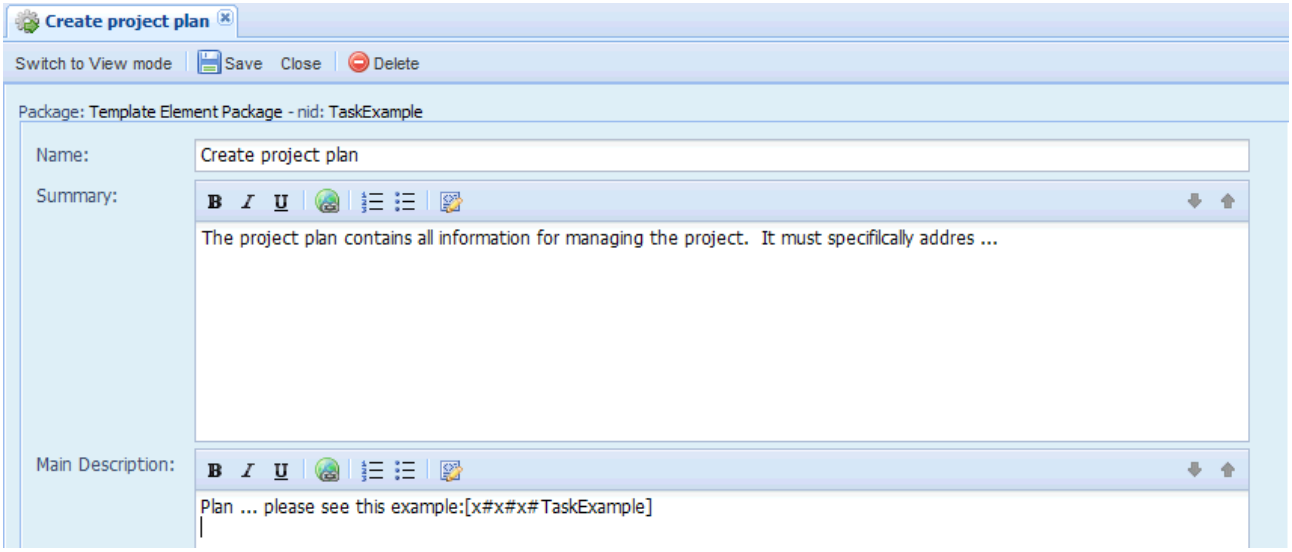
Using the element editor it is possible to edit the data of a specific description element.

New elements are created by right clicking on the relevant type of description element in the “Packages” accordion, see the following picture.



After having entered the name of the element, the element is created below the selected package, and the tree is updated.

Editing an existing description element is done by double clicking the element in the Package view, which opens the element editor. The following picture shows the element editor for a specific task: “Create project plan”.



As shown on the picture, the element editor opens in edit mode, but a preview of the description element will be created when clicking the “Switch to View mode”. Pressing the “Switch to Edit mode” will bring back the editor into the edit mode.

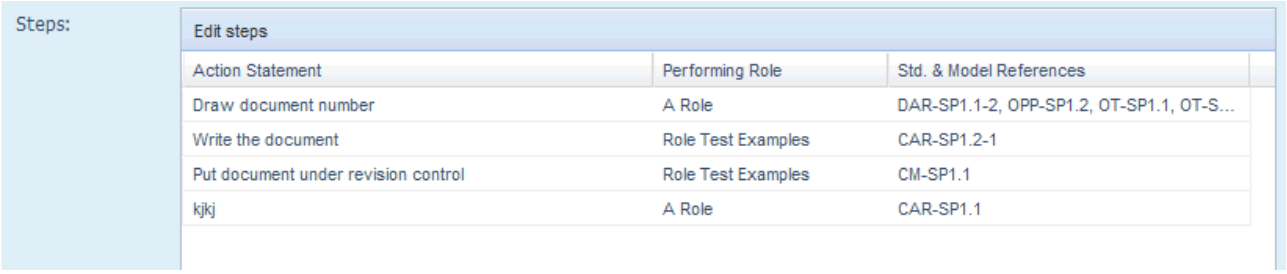
Please notice that in the top line of the element editor, the nid of the element can be seen. When referring to description elements from e.g. MS Visio drawings, this is one of the places where the nid of an element can be found. Furthermore, it is displayed which package the description element belongs to.

After this a number of blocks are presented in the editor, some blocks making it possible to edit text, other specialized blocks making it possible to create references between elements, create steps in tasks etc. The sequence and types of these blocks depends of the actual element edited. Please notice that for most blocks it is possible to drag from the menu trees in the accordions and onto the specific element, creating references and relations the most convenient way.

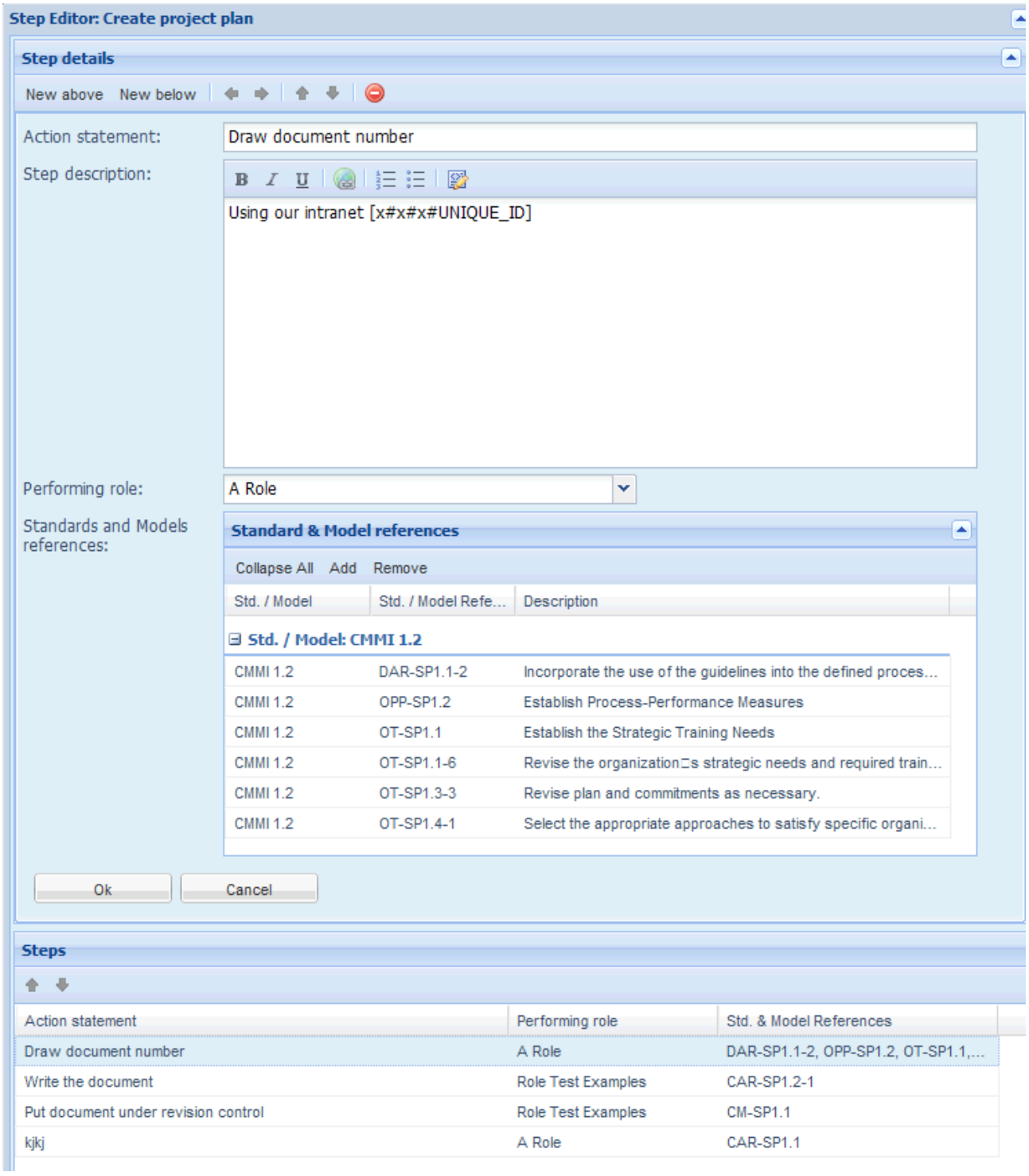
Multiple tabs with element editors can be opened and used at the same time.

4.1.3 The task step editor

One of the components of a "Task" description element is the list of steps to perform in a specific task. When editing a task, the list of steps in that task can be edited by clicking the "Edit steps" button in the top of the "Steps" component, see the following picture of a screen shot of a specific list of steps.



When pressing the “Edit steps” the task step editor appear, see the following illustration.


Step Editor: Create project plan
Step details
New above
New below
Action statement:
Draw document number
Step description:
Using our intranet [x#x#x#UNIQUE_ID]
Performing role:
A Role
Standards and Models references:
Standard & Model references
Collapse All
Add
Remove
Std. / Model
Std. / Model Refe...
Description
Std. / Model: CMMI 1.2
CMMI 1.2 DAR-SP1.1-2 Incorporate the use of the guidelines into the defined proces...
CMMI 1.2 OPP-SP1.2 Establish Process-Performance Measures
CMMI 1.2 OT-SP1.1 Establish the Strategic Training Needs
CMMI 1.2 OT-SP1.1-6 Revise the organization□s strategic needs and required train...
CMMI 1.2 OT-SP1.3-3 Revise plan and commitments as necessary.
CMMI 1.2 OT-SP1.4-1 Select the appropriate approaches to satisfy specific organi...
Ok
Cancel
Steps
Action statement
Performing role
Std. & Model References
Draw document number A Role DAR-SP1.1-2, OPP-SP1.2, OT-SP1.1,...
Write the document Role Test Examples CAR-SP1.2-1
Put document under revision control Role Test Examples CM-SP1.1
kjkj A Role CAR-SP1.1

The task step editor is a “master / detail” list showing first the details of the selected step in the task, and last the master view listing the steps in the task. In this example the “Draw document number” in the master view is selected, which makes the detailed view showing the corresponding information.

Steps can be added before and after the currently selected step. The step currently selected can be moved using the arrows to the left and right. The position/sequence of the steps can be changed by pressing the up and down arrows. The selected step can be deleted by pressing the “stop-sign”.

Please notice that references other elements can be made by dragging from the menu trees in the accordions similar to editing other HTML fields. Furthermore, please notice that a mapping from each of the steps to multiple places in standards and models can be maintained. Using this mechanism, Callis Author will automatically generate the “opposite” mapping (from the standards to the standard process).

Please notice that your changes must be concluded by pressing the “Ok” button in the bottom of the detailed view. This will update the task step editor and save the changes – but only to a buffer. The changes are not committed finally, before save is pressed in the task editor.

4.1.4 The external editor

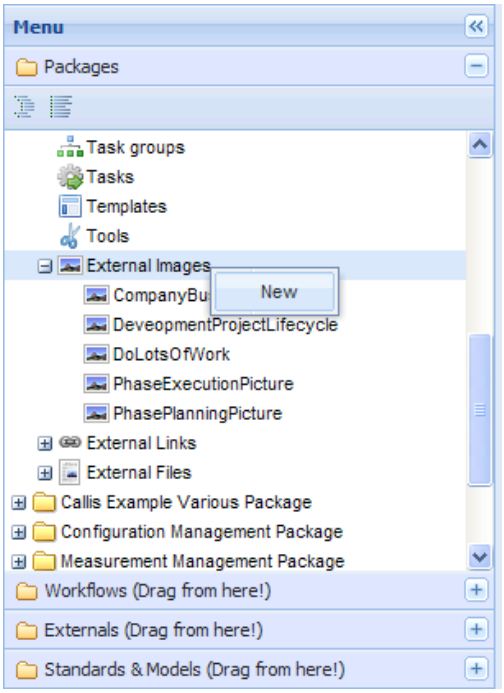
In order to reference an http link, a file, or an image from inside the model you must declare it as an external resource. All resources have a name which is used in the editor and when the elements or links are rendered in the process output.

External resources come in 3 varieties

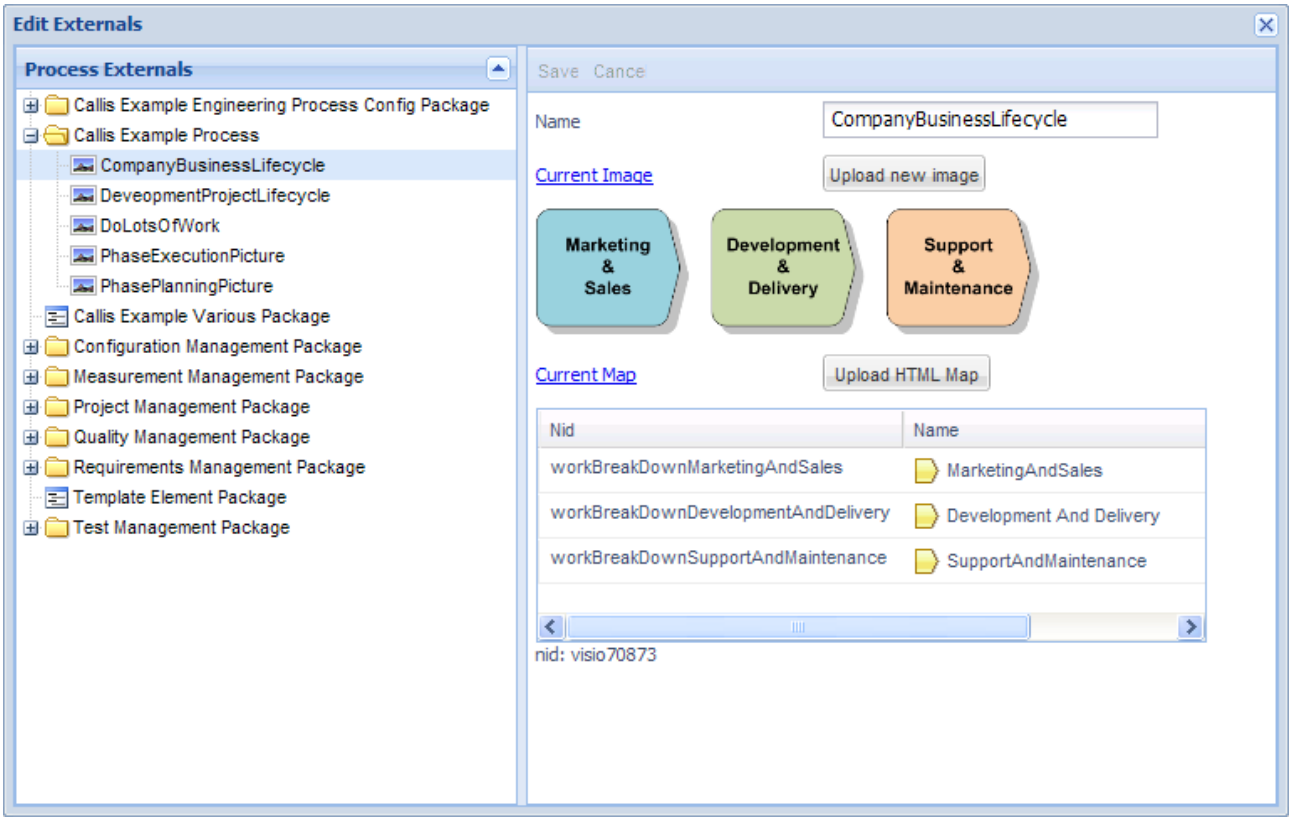
- **link**: This should be used for regular http links. The links can point to e.g. websites or document templates.
- **file**: This is for uploading files including templates, examples, icons etc.
- **image**: This is for uploading images and image-maps.

If an image map is uploaded and the areas in the map have an HREF attribute whose content is on the form of a hash mark (#) followed by a nid of an description element in Callis Author, the images will then be rendered as clickable maps that can be used to navigate the process.

Externals are created and updated using the external editor. New externals can be created by right clicking on one of the three externals groups (files, links or images) below a specific package in the package accordion.



Editing an already existing external can be done by double clicking the external in the packages accordion. This will open the Edit Externals window, see the following picture.



The external editor displays the packages of the standard process. If the packages are opened, the externals are represented by one node for each external. Clicking on a specific external in the menu structure, will display the corresponding data in the panel to the right. On the screen shot above, an external of the type image is displayed. The name of the external can be changed and new image and map file can be uploaded. Please notice that the map file is parsed when uploading and if references to description elements in Callis Author exist, these will be listed in the table. This feature enables the process authors to quickly identify if links are defined correctly on pictures.

4.2 Versioning, approving, and archiving description elements

Callis Author supports versioning, approving, and archiving of description elements.

4.2.1 Versioning mechanism

Callis Author maintains a version number for each description element. The version number uniquely identifies the version of the element and is updated every time the element is saved.
When a description element is saved, the following information is stored:

- Version number
- Name of the user, who changed the element (if available)
- Date and time when the change happened, and
- Note describing the change (if provided by the user)

Please notice, that the previous version of a description element will be overwritten, when a save of that particular description element occurs. Callis Author maintains only the current version of an element (but elements can be explicitly archived and retrieved).

The version information of elements is used to generate standard process versioning information, e.g. when building or deploying standard processes. Furthermore, the versioning information is stored as part of the description element data and is therefore preserved, when copying packages (which includes copying the description elements included in the packages).

4.2.2 Approval mechanism

Description elements have a state (draft, ready, approved). The state system is used to support the approval of description elements.

- **Draft:** The state will automatically change to draft when a save is performed on a description element
- **Ready:** Used to signal from a process reader to the approver of the process element that the element is ready for approval
- **Approved:** Used to show that the description element is approved

The state of a description element can be changed by selecting the “Change state” button in the editor. This will open a window where the new state can selected. Please notice that when saving a description element, automatically the state is changed to draft.

Furthermore, state change operations can performed on a group of description elements by right clicking on the relevant package. This allows for approving multiple descriptions in a single operation. This will open a window where it is possible to select which elements in which states will be changed. Furthermore, the new/end state is selected.

Please notice that changing state does not affect the version of the description element.

4.2.3 Archiving mechanism

Saving and retrieving older versions of description element requires use of the archive/retrieve mechanism. This mechanism allows archiving the current version of a description element and later retrieving it from the archive.

When retrieving the element, the data from the description element is read into a buffer in the editor, where it can be reviewed. If the process author decides that the element is useful, he can save the content as the newest version of the element.

4.3 Editing html blocks

Editing text blocks

4.3.1 Creating links in text blocks

In the fields that support XHTML you can use the construct [x#x#x#nid]. During the compilation, that construct will be expanded to a link to the object with @nid equal to the given value. The x'es are currently just placeholders, but will in later versions of Callis Author be used to configure the rendering of the links.

References can be made either by writing the [x#x#x#nid]-format manually, or by dragging elements from the trees in the accordions onto the HTML edit fields.

Important: Before dragging an element you must click in the HTML field and place the cursor at the position where you want the reference inserted. This controls where the reference is inserted.

4.3.2 Embed clickable images into html fields

If a reference ([x#x#x#nid]) to an external resource of type image is made inside a HTML field (by dragging the relevant external from the references box), then the image will be rendered. This can be verified by saving the element and viewing the element in the Presenter.

4.3.3 What subset of HTML tags are allowed in the Online Editor

The Short answer is: **br,b,i,p,ol,ul,li,div,table,td,tr,th,thead,tbody, h1, h2, h3**

The longer answer is that any element is allowed, but only the ones mentioned will survive the "scrubbing" on the server. Also, even if an element is allowed almost all attributes are stripped during scrubbing. The exception to this is this list of attributes: **rowspan,colspan,align and valign**.

Scrubbing is necessary because we need to render to both html and PDF.

Scrubbing has two steps

1. Convert the HTML to valid XHTML
2. Make it possible to render the elements consistently when browsing, by removing unwanted elements and attributes

Example:

```
This is a <span class="xxx"">example of</span> some list
<ol>
   <li>item 1
</ol>
```

Will after scrubbing be

```
This is a example of some list
<ol>
   <li>item 1</li>
</ol>
```

4.4 Working with pictures

4.4.1 How do I make links from the Visio overview pictures to description elements?

In Visio you select the figure you want to become a hyperlink - press Ctrl-K - and as link you must enter hashmark '#' followed by the nid of the element you want to link to.

E.g if you have a task element with nid attribute value "`mytaskID`" the you should enter:
`#mytaskID`

You must save the Vision file as webpage and upload the image and map file to Callis Author, before the picture can be used in Callis Author, see next section.

4.4.2 How do I insert a Visio picture in an taskgroup or guidance element

Do the following:

1. Open the `mypicture.vsd` in Visio and choose File->Save as webpage and press the OK button. This will create a file called `mypicture.htm` and a directory called **`mypicture_files`** in the same directory as `mypicture.vsd`
2. Start the editor and choose File->Edit Externals
3. Find the package where you want the files to be stored, right-click on the “images” and chose "New " and “OK”.
4. Change the name from "DESCRIPTION" to "My Activity Overview"
5. Pres the "Upload image" button , browse to the **`mypicture_files`** directory and select the `gif_1.gif` and upload it
6. Pres the "Upload map" button , browse to the **`mypicture _files`** directory and select the `gif_1.html` and upload it
7. Open the relevant description element in the editor and select the name of your picture in the overview picture field in the element editor.

8. Save the element
9. Press “Switch to view mode” or browse to the element in the presenter. There should now be a "Overview"-tab containing the uploaded image

If the visio file has more than one drawing i.e. it has multiple tabs then there will be generated multiple pairs of files like gif_1.gif gif_1.html, gif_2.gif gif_2.html etc. Select the right combination when uploading.

4.4.3 The gif files exported by Visio have a very low resolution

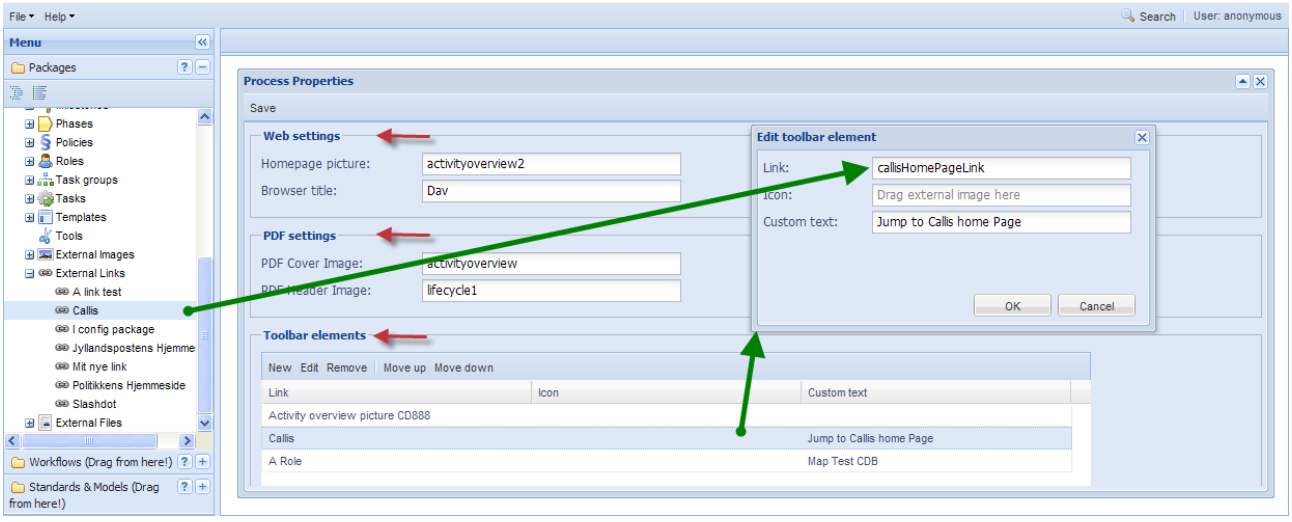
In Visio when you select "Save-as-webpage" there is a button called "Publish". When you press that button and select the "Advanced" tab you can choose the target monitor size. This governs the size of the generated gif files.

4.5 Configuring the publishing properties

The following publishing parameters can be configured as a part of the process:

- Web settings:
  - Home page picture – to specify the home page picture to show when users are accessing the process in Callis Presenter (just drag an image element from the menu tree to this field to specify)
  - Browser title – to specify the title of the tab of the Presenter
- PDF settings:
  - PDF Cover Image – to specify the image to put on the front page of the generated PDF file
  - PDF Header Image – to specify the image to use in the headers of the pages of the PDF file
- Toolbar elements:
  - A list of elements to put in the tool bar

Please notice that all elements can be dragged from the menu tree accordions.



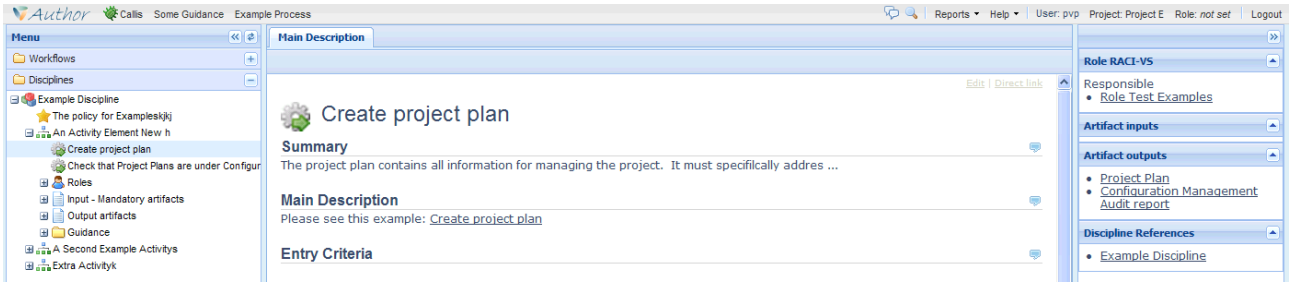
4.6 Publishing the process

Publishing processes is done from Callis Author – Process Control Center. Please refer to section 3.5 to understand how publishing is performed.

5 Browsing a standard process

5.1 Layout of Callis Author - Presenter

The output generated from Callis Author is a process manual which can be browsed. This view is referred to as the Presenter. The following picture shows a screen shot of the Presenter.



In the top bar a number of short cuts are defined (consult Callis Author Administration Guide on how to do this). Furthermore, search button, drop-down menus with reports and help, project and role context and logon button is in the top bar.

In the left hand side a menu consisting of a number of accordions is presented. The accordions represent different views on the standard process currently displayed.

In the main page the selected description element is displayed.

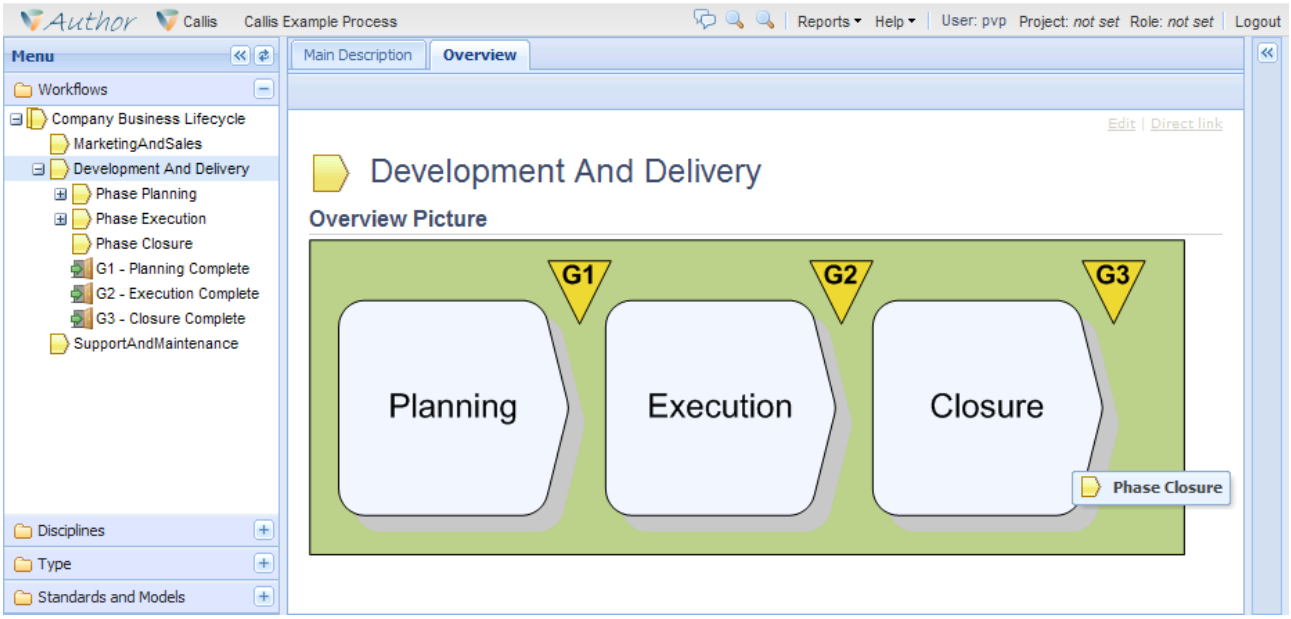
In the right side relations between the selected description element and other description elements are listed.

5.2 Which views exists in the Presenter and how are they generated?

5.2.1 Workflows view

The workflows view is generated based on the following rules:

1. If a workflow element (Lifecycle, Phase, Iteration or Activity) has the attribute “appear as Root element in menu” selected, the element will be listed as a root element in the menu tree below the workflows accordion. In Picture 3 the element ”Company Business Lifecycle” has the attribute set to true.
2. If a workflow element has an overview picture with references to other workflow elements, milestones, gates, or tasks, these elements are listed below that element in the workflow tree, see Picture 3.



Picture 3 illustrates how the tree menu is build automatically based on the content of the uploaded picture.

Using this mechanism, it is very simple and efficient to generate nice process oriented views on top of the method elements created in Callis Author.

5.2.2 Disciplines view

The discipline view uses the relations between disciplines, task groups and tasks defined by the work method elements, see Picture 1. The view is generated by moving from disciplines, taking out the policy, listing the task groups, the tasks below these, and using the relations to roles, artifacts and guidance elements.

5.2.3 Types view

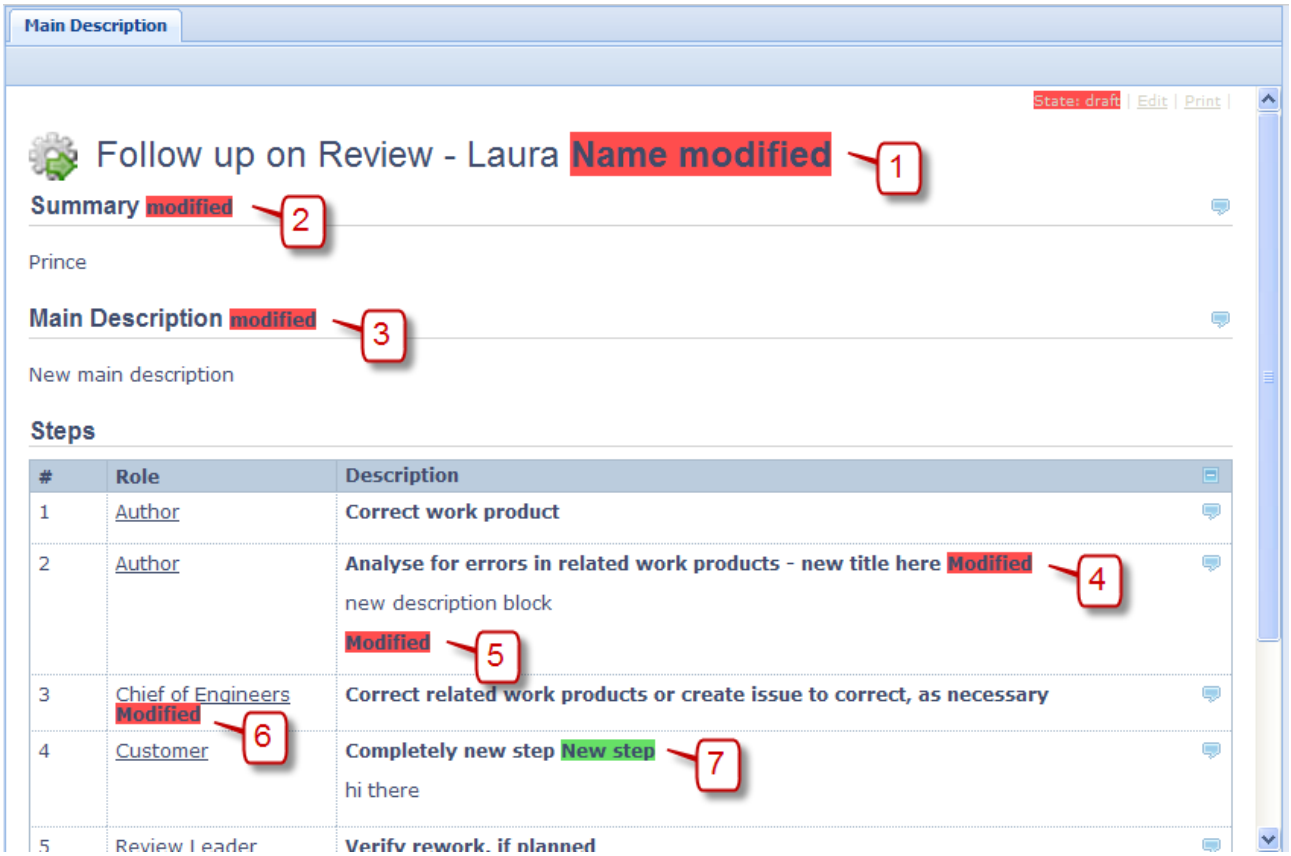
The types view lists all elements sorted first by type of element and then alphabetically.

5.2.4 Standards & Models view

The Standard & Models view shows all standards and models compiled into the standard process. If elements in the standard process are mapped to the standards, Callis Author generates and presents the opposite view (from model to standard process) automatically.

5.3 Change / comparison information

When a process is compared with another process, e.g. an earlier version, changed description element will be marked up with clange information. Modifications are marked with red text, additions are marked with green. The following screen shot illustrates the concept.



The following items refer to the numbers pointing to the change information on the screen shot:

1. Name of description element is modified
2. Summary block is modified
3. Main description block is modified
4. Step action statement / One-liner in the step is modified (but not role or text)
5. Step text is modified
6. Performing role is modified
7. A new step is added

If any modification is done (In the example above from item 1 to 6) the description element will be marked in the change report (please see 6.11). Also new and deleted description elements will be listed in the report (not illustrated by this example).

6 About the reports

When using the reports please notice the following:

- Not all reports are generated run-time (as the processes are changed). These reports are only generated when publishing the standard process. An example of this is the PDF-report with the complete process or discipline PDF reports.
- Where flash charts are used, these can be rendered as PNG-files by pressing the "Open Chart as PNG" button". This will open a browser window with a PNG version of the chart. The PNG image can be copied and then pasted directly into office documents or emails.

- Some reports have links directly to elements in the process. When clicking these links, the tab or window in your browser where Callis Author is running, will update and show the element. Please notice, that the element is not showed in the window where the link is clicked. This is "by purpose", because it makes it easier to quickly walk over a number of links without reloading the report.

6.1 Entire Process as PDF (static, updated when building or publishing)

Description:	PDF document with the entire process. Activities and tasks are listed below disciplines. Other elements alphabetically sorted based on their type.
Useful when:	A printable version of the process is needed or if someone without access to the web version wants to see the process description.
Please notice:	This report is generated only when starting up Callis Author.

6.2 PIID Report template

Description:	An html report showing the mapping of direct and indirect artifacts against the CMMI specific and generic practices.
Useful when:	When an organization wants to collect data for an appraisal, this report is useful to hand out to the people who should provide the evidences.
Please notice:	This is a CMMI specific report, and it does not necessarily make sense for organizations who do not work with the CMMI.

6.3 Statistics

Description:	This html report generates indicators (pie and bar charts) on top of the data showed in the "statistics – Tables"-report. The data is displayed next to the charts.
Useful when:	You want a visual overview of the distribution of element types and steps in tasks.
Please notice:	Copying the numbers to a spreadsheet and comparing the number of description elements and tasks over time will give the organization an idea of how long time it takes to write processes for at specific area.

6.4 Process Integrity

Description:	This model validates the standard process against a number of validation points The validation points are: • **Number of tasks with no CMMI references** list tasks that are not mapped to any standards and models. • **Number of tasks with no output artifacts** lists tasks which have no relation to one or more artifacts (no output is generated). • **Number of tasks with no RACI relationships** lists tasks which have no RACI relation defined (i.e. no-one is responsible for executing the task). • **Number of tasks with 0 steps** lists tasks without any steps defined. • **Number of artifacts that does not appear as output of any task** lists artifacts which have no output (aka generated by) relation to a task.
Useful when:	Checking if all relevant relations are created in a standard process.

Please notice:	• The report lists the number of elements detected (the number in blue to the right). When clicking the number, it lists links to the elements detected. Then pressing one of the links, Callis Author updates to show that exact element. • Description elements that are detected against these validation points are not necessarily modeling errors, but “things to consider”.

6.5 Coverage against Standards

Description:	This report traverses all standards, and for all entries in the standards, the mapping to the standard process is listed.
Useful when:	• Checking if all parts of a standard have been mapped to the standard process •
Please notice:	When clicking on the links in the right part of the report, Callis Author updates to show that exact element.

6.6 Process Overview

Description:	This report generates an overview of the architecture of the standard process.
Useful when:	• Getting an overview of the disciplines, activities and tasks • Understanding how a task relates to artifacts and roles • Getting an quick overview of the steps in a task
Please notice:	

6.7 Who references [current]

Description:	This report generates shows an overview of all elements which have references to the current element.
Useful when:	• Understanding where a specific description element is referred
Please notice:	

6.8 Unresolved References

Description:	This report detects all references that cannot be resolved and lists the description elements from where the reference is made.
Useful when:	Combining packages to new standard processes and checking if all elements referenced are included somewhere.
Please notice:	If a package is used in more than one standard process, be careful when fixing the references. The fix should work of all standard processes generated based on this package.

6.9 Element versions – Grouped by package

Description:	This report lists which description elements in which versions are included in the standard process.
Useful when:	Documenting the configuration of a standard process
Please notice:	

6.10 Element versions – Grouped by save date

Description:	This report lists description elements with descending modified date. The report is grouped on packages.
Useful when:	Anyone wants to know if any changes have been made on a specific part of the process.
Please notice:	Only the latest changed date is listed of each description element.

6.11 Elements compared to other process

Description:	This report generates an overview of differences between two processes.
Useful when:	• Generating change logs (from previous to current released version of the process) • Getting an overview of the tailorings performed in a specific project
Please notice:	The report will list new(added), modified and deleted description elements. When clicking on one of the links in the report, the presenter will show the element with the changes marked in the element text. Please notice that it is not possible to display deleted elements – these are only listed in the report. A process definition refers to a set of packages. If a process definition is made including a specific version of a package e.g. ReqmPackage v. 1.1 and another process is referring the same package, change the package content in one of the processes will change the content in the other process also. In this case there will be no differences between the process definitions. If you want to be able to detect changes between different versions of processes, you need to copy the relevant packages (and thereby create a baseline).

7 Comments

Callis Author provides you with a simple and lightweight commenting mechanism. The intention is a mechanism useable for easy review and commenting on a process description.

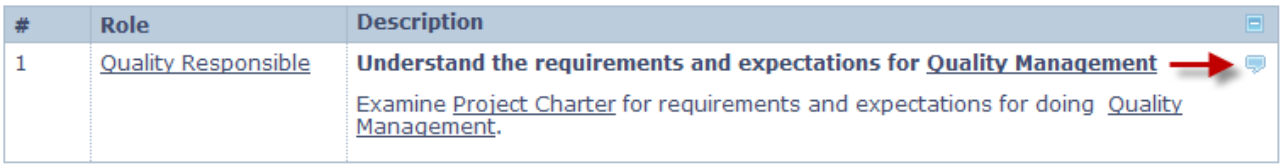
Please notice that commenting does not work on published processes as these are static.

7.1 Commenting on published process descriptions

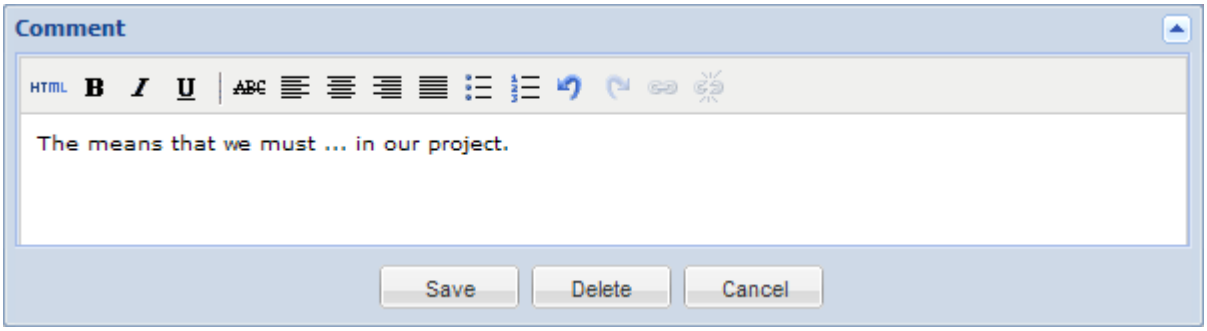
Comments are used to do lightweight review or commenting of the process description to fit the need of specific projects. Comments are always done in the scope of a project.

7.1.1 Do a comment

Just press the comments icon in the right, top corner of the relevant text block. Remember that a project must be selected before commenting.



Clicking the icon will open the comments window for your specific project:



You can now enter a comment which will go into the standard process as a project specific tailoring. Please see the following picture to see how that works.

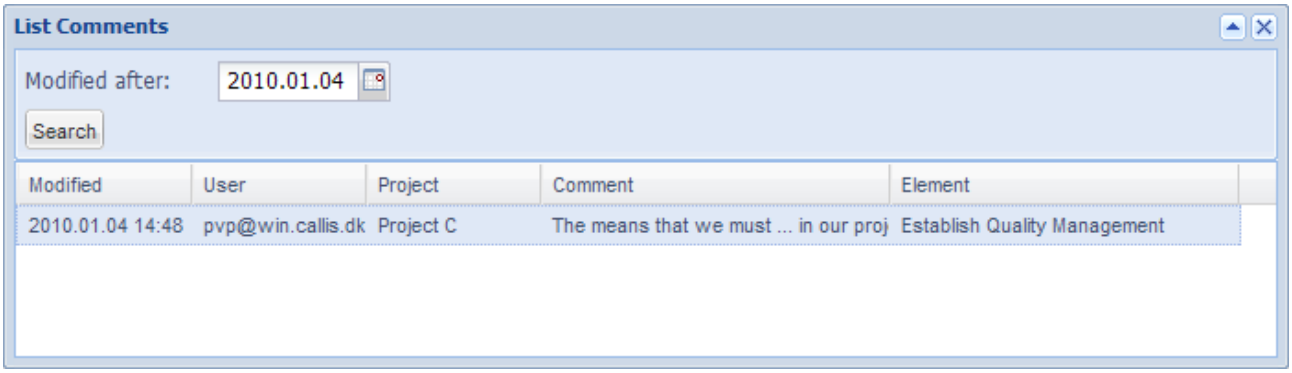
#
Role
Description
1
Quality Responsible
Understand the requirements and expectations for Quality Management
Examine Project Charter for requirements and expectations for doing Quality Management.
The means that we must ... in our project.
Project C | 2010-01-04 13:48:06 by pvp@win.callis.dk

Comments are a light weight and easy to use mechanism to make the process description fit the need of a specific project - without requiring competence to actually re-model the process description.

7.1.2 Get an overview of recently changed comments

A project can get an overview of the comments made for their project. This is done by selecting the comments menu in the toolbar and selecting the “list comments” menu item.

The “list comments” will show a table where each row represent a comment. The entries in the table are filtered based on the date the comments were submitted.



Clicking on one of the rows will make the Presenter scroll to the description element with the specific comment and flash the comment.

Please notice, that in general this mechanism is intended to make project process responsible and project managers keep track of the comments made to the project specific process.

7.1.3 See comments across a set of projects

It is possible to use the “list comments” mechanism to show comments across multiple projects. In the comments menu, select the “Configure comments across projects” menu item. This will open a window with two panels. The panel to the left contains the hierarchy of projects defined. The panel to the right contains the projects for which comments are displayed. Double click on a node to move it from one side to the other.

Please notice that comments will always be shown for the logged in user's current project.

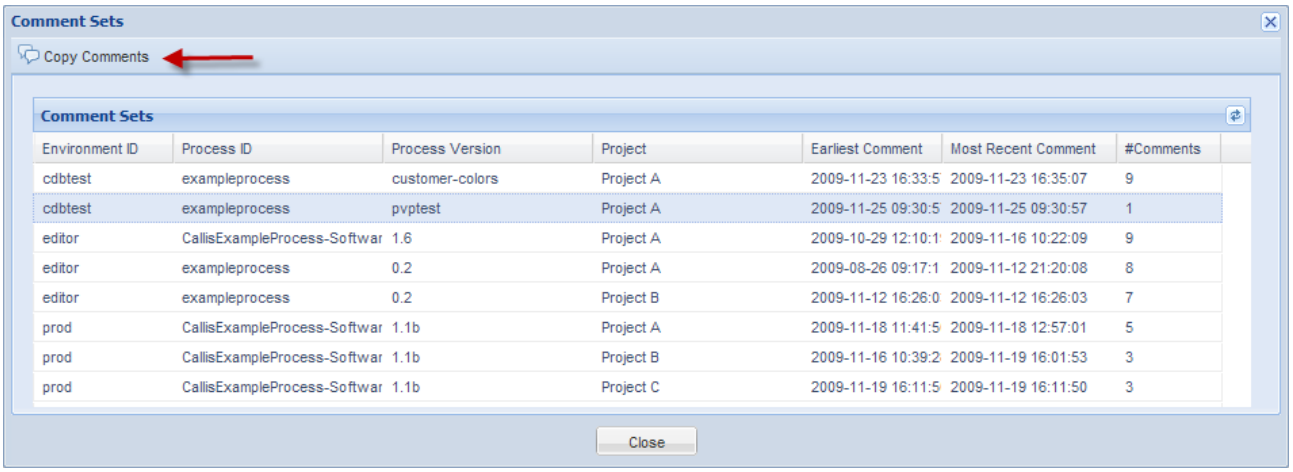
7.1.4 Migrate comments across process versions

Comments are stored in Callis comments database using the following attributes forming a unique key:

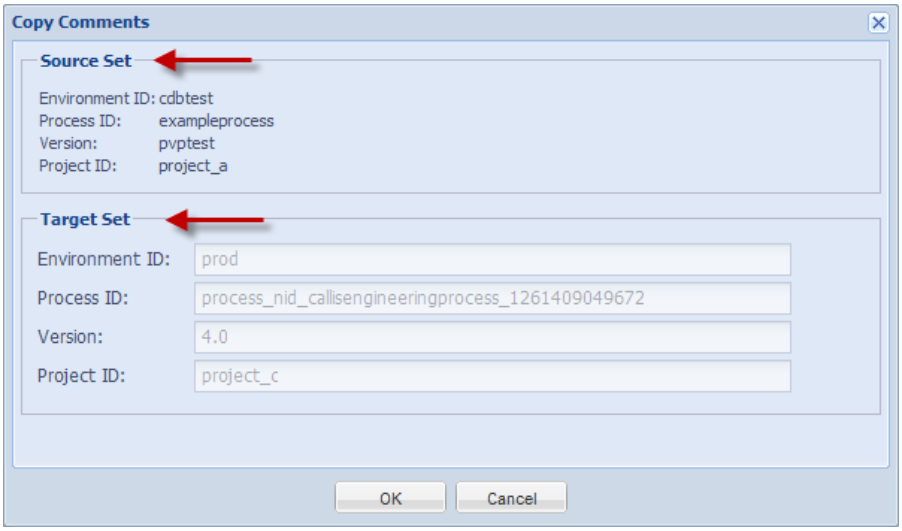
- **The following attributes define a comment set:**
  - **Environment** is defined by which environment the process is published to.
  - **Process ID** is defined in Callis Author – Process Control Center when creating a process definition. You can find the ID of a specific process definition as the “Nid”
  - **Process Version** is defined in Callis Author – Process Control Center when defining or modifying a process definition
  - **Project ID** is defined in Callis Author – Administration when defining projects.
- **The following attributes defines where to put the comment**
  - **Description element ID** is an id uniquely defining a specific description element. You can see the description element ID

- **SubID** defines which block on a specific element the comment is attached to, e.g. “main description”, “summary”, or “step 2”.

Comments in a comment set can be copied to the process currently browsed. Just select the relevant comments set from the “Copy comments” window (find this in the “comments” menu).


Comment Sets
Copy Comments
Comment Sets
Environment ID
Process ID
Process Version
Project
Earliest Comment
Most Recent Comment
#Comments
cdbtest exampleprocess customer-colors Project A 2009-11-23 16:33:5 2009-11-23 16:35:07 9
cdbtest exampleprocess pvptest Project A 2009-11-25 09:30:5 2009-11-25 09:30:57 1
editor CallisExampleProcess-Softwar 1.6 Project A 2009-10-29 12:10:1 2009-11-16 10:22:09 9
editor exampleprocess 0.2 Project A 2009-08-26 09:17:1 2009-11-12 21:20:08 8
editor exampleprocess 0.2 Project B 2009-11-12 16:26:0 2009-11-12 16:26:03 7
prod CallisExampleProcess-Softwar 1.1b Project A 2009-11-18 11:41:5 2009-11-18 12:57:01 5
prod CallisExampleProcess-Softwar 1.1b Project B 2009-11-16 10:39:2 2009-11-19 16:01:53 3
prod CallisExampleProcess-Softwar 1.1b Project C 2009-11-19 16:11:5 2009-11-19 16:11:50 3
Close

When pressing the “copy comments” button, the following window will appear:



This will bring up a window presenting key attributes in both Source Set and Target Set of comments. Just press ok, and the comments are copied to the current process description.

Please notice the following:

1. When a new version of a process is published, projects need to copy existing comments from the previous version to the new version to see their comments in the new process
2. If the process authors have removed parts of the process description, there is a risk that some comments will not be rendered when browsing the new process description. This will happen if a

comment was attached to a part of the process which is removed in the new version. No warnings will be given in this case.